All pending claims of the present Application are shown below.

1. (Previously Presented) A method for preparing files for storage in a server comprising:

generating a profile for a selected file, the profile identifying at least one associated file to be accessed by the selected file; and

transmitting, to a server, the selected file, the profile, and the at least one associated file.

2. (Original) The method of Claim 1, and further comprising:

after transmitting the selected file, the profile, and the at least one associated file, initiating downloading of the selected file from the server;

identifying the at least one associated file by examining the profile; and

in response to identifying the at least one associated file by examining the profile, initiating downloading of the at least one associated file from the server.

3. (Original) The method of Claim 1, wherein the profile identifies the at least one associated file using a Uniform Resource Locator.

4. (Original) The method of Claim 1, and further comprising associating a globally unique identifier with each of the files, wherein the profile additionally identifies the at least one associated file by the respective globally unique identifiers.

5. (Original) The method of Claim 4, and further comprising:

after transmitting the selected file, the profile, and the at least one associated file, determining if any of the at least one associated file is a missing file, wherein the missing file is any of the at least one associated file that has a different identifier than the identifier used by the profile to identify the at least one associated file; and

searching, using a globally unique identifier associated with each of the at least one associated file, for the missing file.

6. (Original) The method of Claim 5, and further comprising:

determining the different identifier of the missing file; and

updating the profile with the different identifier for the missing file.

7. (Original) The method of Claim 1, and further comprising:

receiving, at the server, the selected file, the profile, and the at least one associated file; and

indexing, at the server by a document manager residing in the server, the profile.

8. (Original) The method of Claim 2, and further comprising, in response to initiating downloading of the at least one associated file from the server, storing the at least one associated file in a memory associated with a client under a local identifier.

9. (Previously Presented) A method for file management, comprising:

generating, at a client device, a profile for a selected file that is to be downloaded from a server, the profile identifying all associated files to be accessed by the selected file after the selected file is downloaded from the server;

transmitting, to a server, the selected file, the profile, and all of the associated files;

after transmitting the selected file, the profile, and all of the associated files, initiating downloading of the selected file from the server;

identifying all of the associated files by examining the profile; and

in response to identifying all of the associated files, initiating downloading of all of the associated files from the server.

10. (Original) The method of Claim 9, wherein the profile identifies the at least one associated file using a Uniform Resource Locator.

11. (Previously Presented) The method of Claim 9, and further comprising associating a globally unique identifier with each of the files, wherein the profile additionally identifies at least one of the associated files by the respective globally unique identifiers.

12. (Previously Presented) The method of Claim 11, and further comprising:

after transmitting the selected file, the profile, and all of the associated files, determining if any of the associated files is a missing file, wherein the missing file is any of the associated files that has a different identifier than the identifier used by the profile to identify the associated file; and

searching for the missing file using the globally unique identifier of the missing file.

13. (Original) The method of Claim 12, and further comprising:

determining the different identifier of the missing file; and

updating the profile with the different identifier for the missing file.

14. (Original) The method of Claim 9, and further comprising:

receiving, at the server, the selected file, the profile, and the at least one associated file; and

indexing, at the server by a document manager residing in the server, the profile.

15. (Previously Presented) The method of Claim 9, and further comprising, in response to initiating downloading all of the associated files from the server, storing the associated files in a memory associated with a client using a plurality of local identifiers.

16. (Original) A method for preparing a plurality of files for storage in a server comprising:

providing a parent file having at least one level of descendent files;

generating a profile for the parent file identifying all of the descendent files that are immediately associated with the parent file as immediately associated with the parent file;

for each level of the descendent files, generating a profile for each descendent file in the level, the profile identifying all of the descendent files that are immediately associated with the descendent file as immediately associated with the descendent file; and

transmitting the parent file, each descendent file in each level of the descendent files, and the profiles to the server.

17. (Original) The method of Claim 16, and further comprising:

after transmitting the parent file, each descendent file in each level of the descendent files, and the profiles, initiating downloading of the parent file from the server;

identifying the descendent files in each level of the descendent files by examining the profiles; and

in response to identifying the at least one associated file, initiating downloading of all of the descendent files in each level of the descendent files from the server.

18. (Original) The method of Claim 17, and further comprising compiling a list of all descendent files, and initiating downloading of all of the descendent files identified on the list.

19. (Original) The method of Claim 16, wherein the profile identifies the each of the descendent files in each level of the descendent files using a Uniform Resource Locator.

20. (Original) The method of Claim 16, and further comprising associating a globally unique identifier with each of the plurality of files, wherein the profile additionally identifies each descendent file by the respective globally unique identifiers.

21. (Original) The method of Claim 17, and further comprising:

after transmitting the parent file, the each descendent file in each level of the descendent files, and the profiles to the server, determining if any of the each descendent file is a missing file, wherein the missing file is any of the each descendent file that has a different identifier than the identifier used by the profile to identify the each descendent file; and

searching, using a globally unique identifier associated with the each descendent file, for the missing file.

22. (Original) The method of Claim 21, and further comprising:

determining the different identifier of the missing file; and

updating the profile with the different identifier for the missing file.

23. (Original) The method of Claim 16, and further comprising:

receiving, at the server, the parent file, the profiles, and the descendent file; and

indexing, at the server by a document manager residing in the server, the profile.

24. (Original) The method of Claim 17, and further comprising,

in response to initiating downloading of all of the descendent files in each level of the descendent files from the server, storing the all of the descendent files in each level of the descendent files in a memory associated with a client under a local identifier.

25. (Previously Presented) An apparatus for preparing files for storage in a server comprising:

software stored on a computer readable medium and operable to:

generate a profile for a selected file, the profile identifying at least one associated file to be accessed by the selected file; and

initiate transmission, to a server, of the selected file, the profile, and the at least one associated file.

26. (Original) The apparatus of Claim 25, wherein the software is further operable to:

initiate downloading of the selected file from the server;

identify the at least one associated file by examining the profile; and

in response to identifying the at least one associated file, initiate downloading of the at least one associated file from the server.

27. (Original) The apparatus of Claim 25, wherein the software comprises a drawing package.

28. (Original) The apparatus of Claim 25, wherein the profile identifies the at least one associated file using a Uniform Resource Locator.

29. (Original) The apparatus of Claim 25, wherein the software is further operable to associate a globally unique identifier with each of the files, wherein the profile additionally identifies the at least one associated file by the respective globally unique identifiers.

30. (Original) The apparatus of Claim 30, wherein the software is further operable to:

after transmitting the selected file, the profile, and the at least one associated file, determine if any of the at least one associated file is a missing file, wherein the missing file is any of the at least one associated file that has a different identifier than the identifier used by the profile to identify the at least one associated file; and

search, using a globally unique identifier associated with each of the at least one associated file, for the missing file.

31. (Original) The apparatus of Claim 29, wherein the software is further operable to:

determine the different identifier of the missing file; and

update the profile with the different identifier for the missing file.

32. (Original) The apparatus of Claim 26, wherein the software is further operable to, in response to initiating downloading of the at least one associated file from the server, store the at least one associated file in a memory associated with a client under a local identifier.